**What is claimed is:**

1. A method of compressing source image data for a hard copy device, the method comprising:

storing the source image data in units of bytes, in a memory in rows and columns;

transposing bytes at each column of the source image data to bytes at each row of the source image data;

determining start information, said start information containing header information and containing information on a chain that was the starting point where compression on the source image data is started;

encoding a plurality of sequential chains and generating compressed data including said encoded chains;

each one of said sequential chains corresponding to one selected from among a current chain alone and the current chain with at least one chain succeeding the current chain, said selecting being in dependence upon whether a dictionary contains chains having same value as the current chain, the dictionary being composed of chains preceding the current chain; and

including said start information in said compressed data, each chain being composed of at least two consecutive bytes at a row, neighboring bytes at each row having neighboring memory addresses, offset of neighboring bytes at each column corresponding to width of row of the source image data.

2. The method of claim 1, said encoding and generating further comprising:

reading value of the current chain, said read value corresponding to a first value;

determining whether a matched chain having said first value exists in the dictionary;

when at least one matched chain exists in the dictionary, performing a first process of obtaining a first count value by counting number of said sequential chains having values which match values of chains in the dictionary, and encoding value of *i* of the at least one matched chain and the first count value, *i* being an index indicating each chain included in the dictionary;

when a matched chain does not exist in the dictionary, performing a second process of obtaining a second count value by counting number of said sequential chains not having values which match values of chains in the dictionary, and storing the value of the current chain;

after performing one selected from among said obtaining of said first count value and said obtaining of said second count value, determining whether the current chain is the last chain to be compressed;

when the current chain is not the last chain, setting a chain succeeding the current chain to be a new current chain and performing said reading of value of the current chain;

when the current chain is the last chain, performing a third process of encoding a predetermined initial value and the second count value in dependence upon the second count value; and

synthesizing the start information and the current chain's value stored in said second

process with results obtained in said first and third processes to perform said generating of compressed data.

3. The method of claim 4, the dictionary corresponding to a surround comprising all chains preceding the current chain.

4. The method of claim 2, further comprising:

determining a template comprising chains preceding the current chain, the chains in the template having values selected from among values which are the same as the current chain and values that are similar to the values of the current chain, the dictionary corresponding to the template, said determining of the template being performed after said transposing and before said encoding of the plurality of sequential chains.

5. The method of claim 4, said determining of the template comprising:

obtaining surrounds of respective pseudo-random chains applied to the source image data transposed by said transposing;

counting at least one chain having the same value of each pseudo-random chain in a corresponding surround;

sorting count values by indexes indicating chains included in each surround, said count values being obtained in said counting of the at least one chain having the same value of each

pseudo-random chain in a corresponding surround; and

selecting predetermined number *m* of count values starting from a maximum count value among the sorted count values, *m* being one selected from among being equal to the number of chains in the surround and being less than the number of chains in the surround, each surround being composed of chains which have been compressed before a corresponding pseudo-random chain is compressed, the template being composed of chains indicated by indexes corresponding to the selected count values.

6. The method of claim 5, the start information being related to a chain which is first compressed among the chains included in the template and which has a history, and the history is composed of at least one previous chain.

7. The method of claim 6, among the chains included in the template, only chains whose values *c-T[i]* are not negative are included in the dictionary, *c* indicating the index of the current chain, *T[i]* indicating the distance between the current chain and a chain indicated by *i*, and $1 \leq i \leq m$.

8. The method of claim 5, the start information being related to a chain having a maximum *T[i]* among the chains included in the template, *T[i]* indicating the distance between the current chain and a chain indicated by *i*, and $1 \leq i \leq m$.

9. The method of claim 5, said generating of the compressed data further comprising generating compressed source image data in a data format comprising:

chunk data having one selected from among a first group of information and a second group of information, said first group of information corresponding to *i* and the first count value which are encoded in said first process, said second group of information corresponding to the predetermined initial value and the second count value which are encoded in said third process, the chunk data being one-byte aligned;

chain data having the start information which is not encoded and the value of the current chain which is stored in said second process without being encoded; and

a header having the size of the chain data and the length of the dictionary.

10. The method of claim 9, the chunk data comprising at least *M* bits and 8-*M* bits, the *M* bits indicating one selected from among *i* of the matched chain and the predetermined initial value, the 8-*M* bits indicating one selected from among the first count value and the second count value, relationship among *M* and *m* being shown by $m=2^{M-1}$.

11. The method of claim 10, the chunk data further comprising additional bits indicating only one selected from among the first count value and the second count value.

12. The method of claim 11, *M* being a value selected from among 3, 4, 5, and 6.

13. The method of claim 9, said first process further comprising:

flushing the second count value when the matched chain exists in the dictionary; and

encoding *i* of the matched chain and the first count value.

14. The method of claim 13, said second process further comprising:

determining and storing the value of the current value as the chain data when the matched chain does not exist in the dictionary;

increasing the second count value by 1;

when the second count value is not the predetermined maximum value, determining whether the second count value is a predetermined maximum value; and

when the second count value is the predetermined maximum value, flushing the second count value.

15. The method of claim 14, said third process further comprising:

determining whether the second count value exceeds a predetermined threshold value when the current chain is the last one to be compressed;

when the second count value does not exceed the predetermined threshold value, terminating the data compression; and

when the second count value exceeds the predetermined threshold value, encoding the predetermined initial value and the second count value and terminating the data compression.

16. The method of claim 2, said first process further comprising obtaining the first count value by counting the number of sequential chains having the same value, and run length coding the first count value.

17. The method of claim 1, the dictionary being composed of previous chains compressed before the current chain.

18. The method of claim 1, further comprising hard copying a bilevel screened image using the source image data.

19. The method of claim 1, said encoding further comprising determining result of said encoding.

20. The method of claim 1, said encoding corresponding to entropy encoding.

21. The method of claim 1, the current chain being the chain to be currently compressed.

22. The method of claim 1, each chain comprising two bytes at a row.

23. The method of claim 1, each chain comprising four bytes at a row.

24. An apparatus for compressing source image data, the apparatus comprising:

a first data transposer receiving the source image data, the source image data being stored in units of bytes in rows and columns, said first data transposer transposing bytes at each column to bytes at each row and outputting the result of the transposition;

a template determiner receiving the result of the transposition from said first data transposer and determining a template, the template being composed of previous chains having values selected from among values that are the same as a current chain and values that are similar to the current chain, the current chain being the chain to be currently compressed, the previous chains having been compressed before the current chain is compressed; and

an encoder inspecting the template received from said template determiner to determine whether the template has chains having the same value as the current value received from said first data transposer, said inspecting being performed in response to start information containing information on a chain to be first compressed, said encoder encoding sequential chains composed

of one selected from among the current chain and the current chain with at least one chain succeeding the current chain, said encoding being performed in response to result of said inspecting, said encoder outputting result of said encoding as result of compressing the source image data;

each chain being composed of at least two consecutive bytes at a row, neighboring bytes at each row having neighboring memory addresses, offset of neighboring bytes at each column corresponding to the row width of the source image data;

the start information and header information being predetermined, provided to said encoder, and included in the result of the compression.

25. The apparatus of claim 24, further comprising a memory storing the source image data in units of bytes in rows and columns, said first data transposer receiving the source image data from said memory.

26. The apparatus of claim 24, said apparatus being used for hard copying a bilevel screened image in a hard copy device, the bilevel screened image corresponding to the source image data.

27. The apparatus of claim 24, said encoder corresponding to an entropy encoder, said encoding corresponding to entropy encoding.

28. The apparatus of claim 24, said encoder comprising:

a data input unit reading value of the current chain received from said first data transposer in response to the start information, the header information, and a first control signal;

a first match inspector inspecting the template received from said template determiner to determine whether a matched chain having the same value as the value read by said data input unit exists in the template;

said first match inspector detecting a first match signal indicating one selected from among the existence and the nonexistence of a matched chain, in dependence upon said inspecting;

said first match inspector detecting *i* of the matched chain when a matched chain exists, *i* being an index indicating each chain in the template;

a first counter counting the number of sequential chains having a matched chain in the template in response to the first match signal and outputting the result of the counting as a first count value;

a second counter counting the number of sequential chains, which do not having a matched chain in the template, in response to the first match signal, and outputting the result of the counting as a second count value;

a first encoder encoding the first count value and *i* of the matched chain received from said first match inspector, and outputting the result of the encoding;

a comparator comparing the second count value with a predetermined threshold value in response to the first control signal, and outputting the result of the comparison;

a second encoder encoding a predetermined initial value which is externally input and the second count value in response to the result of the comparison received from said comparator;

a storage unit storing the results of the encoding received from said first and second encoders, the value of the current chain received from said first data transposer, and the start information which is externally input, in response to the first match signal;

a first controller determining whether the current chain received from said first data transposer is the last chain to be compressed in the source image data and outputting the result of the determination as the first control signal; and

a first data output unit synthesizing the start information, the value of the current chain, and the results of the encoding of said first and second encoders which are received from said storage unit and outputting the result of the synthesis as the result of the compression.

29. The apparatus of claim 28, said storage unit comprising:

a first storage unit selectively storing one selected from among the result of the encoding of said first encoder and the result of the encoding of said second encoder as chunk data in response to the first match signal; and

a second storage unit storing the value of the current chain and the start information as chunk data in response to the first match signal.

30. A method of compressing original source image data and then reconstructing the original source image data from the compressed data, the method comprising:

storing the source image data in units of bytes, in a memory in rows and columns;

performing a first transposing process by transposing bytes at each column of the source image data to bytes at each row of the source image data, said first transposing process being performed by a first transposer;

determining start information, said start information containing header information and containing information on a chain that was the starting point where compression on the source image data is started;

encoding a plurality of sequential chains and generating compressed source image data including said encoded chains;

each one of said sequential chains corresponding to one selected from among a current chain alone and the current chain with at least one chain succeeding the current chain, said selecting being in dependence upon whether a dictionary contains chains having same value as the current chain, the dictionary being composed of chains preceding the current chain;

said encoding and generating further comprising:

reading value of the current chain, said read value corresponding to a first value;

determining whether a matched chain having said first value exists in the dictionary;

when at least one matched chain exists in the dictionary, performing a first process of obtaining a first count value by counting number of said sequential chains having values which match values of chains in the dictionary, and encoding value of *i* of the at least one matched chain and the first count value, *i* being an index indicating each chain included in the dictionary;

when a matched chain does not exist in the dictionary, performing a second process of obtaining a second count value by counting number of said sequential chains not having values which match values of chains in the dictionary, and storing the value of the current chain;

after performing one selected from among said obtaining of said first count value and said obtaining of said second count value, determining whether the current chain is the last chain to be compressed;

when the current chain is not the last chain, setting a chain succeeding the current chain to be a new current chain and performing said reading of value of the current chain;

when the current chain is the last chain, performing a third process of encoding a predetermined initial value and the second count value in dependence upon the second count value; and

synthesizing the start information and the current chain's value stored in said second process with results obtained in said first and third processes to perform said generating of compressed data;

determining a template comprising chains preceding the current chain, the chains in the template having values selected from among values which are the same as the current chain and values that are similar to the values of the current chain, the dictionary corresponding to the template, said determining of the template being performed after said transposing and before said encoding of the plurality of sequential chains;

said determining of the template further comprising:

obtaining surrounds of respective pseudo-random chains applied to the source image data transposed by said first transposing process;

counting at least one chain having the same value of each pseudo-random chain in a corresponding surround;

sorting count values by indexes indicating chains included in each surround, said count values being obtained in said counting of the at least one chain having the same value of each pseudo-random chain in a corresponding surround; and

selecting predetermined number *m* of count values starting from a maximum count value among the sorted count values, *m* being one selected from among being equal to the number of chains in the surround and being less than the number of chains in the surround, each surround being composed of chains which have been compressed before a corresponding pseudo-random chain is compressed, the template being composed of chains indicated by indexes corresponding to the selected count values.

said generated compressed data having a data format comprising:

chunk data having one selected from among a first group of information and a second group of information, said first group of information corresponding to *i* and the first count value which are encoded in said first process, said second group of information corresponding to the predetermined initial value and the second count value which are encoded in said third process, the chunk data being one-byte aligned;

chain data having the start information which is not encoded and the value of the current chain which is stored in said second process without being encoded; and

header data having the size of the chain data and the length of the dictionary.

31. The method of claim 30, further comprising reconstructing original source image data from the compressed source image data, said reconstructing comprising:

performing a first extracting process by extracting the header and the start information from the compressed source image data;

performing a first decoding process by decoding compressed chains contained in the chunk data and the chain data which are extracted from the compressed source image data, using the header and the start information extracted by said first extracting; and

performing a second transposing process by transposing bytes at each column to bytes at each row in the result of said first decoding and determining the result of the transposition as reconstructed source image data.

32. The method of claim 31, said first decoding process further comprising:

performing a second extracting process by extracting the chunk data from the compressed source image data using the header and the start information;

determining whether the chunk data extracted in said second extracting has one selected from among the predetermined initial value and the *i* of the matched chain;

when the chunk data has the predetermined initial value, performing a second decoding process by decoding the second count value contained in the chunk data, and reconstructing original chains from the compressed chains which do not have matched chains using the chain data extracted from the compressed source image data and the result of said decoding;

when the chunk data has the *i* of the matched chain, performing a third decoding process by decoding the chunk data and reconstructing original chains from the compressed chains having matched chains using the decoded chunk data;

determining whether the chunk data is the last one in the compressed source image data and proceeding to said second extracting process when the chunk data is not the last one; and

when the chunk data is the last one, synthesizing the chains reconstructed in one decoding process selected from among said second and third decoding processes, to generate the result of said first decoding process.

33. The method of claim 30, further comprising reconstructing original source image data from the compressed source image data, said reconstructing comprising:

extracting the header and the start information from the compressed source image data which is input, said extracting being performed by an information extractor;

entropy decoding compressed chains contained in the chunk data and the chain data which are extracted from the compressed source image data, in response to the header and the start information received from said information extractor, and outputting the result of the entropy decoding, said entropy decoding being performed by an entropy decoder connected to said information extractor; and

performing a second transposing process by transposing bytes at each column to bytes at each row in the result of said entropy decoding received from said entropy decoder, and outputting the result of the transposition as reconstructed source image data.

34. The method of claim 33, said second transposing process being performed by a second data transposer.

35. The method of claim 33, further comprising:

performing said compressing of the source image data in a computer;

transmitting the compressed source image data from the computer to a hard copy device;

performing said reconstructing of the original source image data in the hard copy device; and

printing data on a transparent medium, said data corresponding to the source image data,

said printing being performed by the hard copy device.

36. The method of claim 24, said entropy decoder comprising:

a data extractor extracting the chunk data from the compressed source image data, in response to the header and the start information received from said information extractor and a second control signal;

a second match inspector inspecting the chunk data to determine whether the chunk data extracted by said data extractor has one selected from among the predetermined initial value and the *i* of the matched chain, and outputting the result of the inspection as a second match signal;

a first chain reconstructor decoding the second count value contained in the chunk data in response to the second match signal, reconstructing original chains from the compressed chains which do not have matched chains using the chain data extracted from the compressed source image data and the result of said decoding, and outputting the reconstructed chains;

a second chain reconstructor decoding the chunk data in response to the second match signal, reconstructing original chains from the compressed chains having matched chains using the decoded chunk data, and outputting the reconstructed chains;

a second controller determining whether the chunk data received from said data extractor is the last one in the compressed source image data, and outputting the result of the determination as the second control signal; and

a second data output unit synthesizing the reconstructed chains received from said first

and second chain reconstructors in response to the second control signal, and outputting the result of the synthesis as the result of said entropy decoding.